Early Experiences with Energy-Aware Scheduling

Kathleen A. Smith, kathleen.a.smith120.civ@mail.mil, *ARL DSRC*
William DeSalvo, bdesalvo@instrumental.com, *Instrumental*
Michael P. Knowles, michael.p.knowles8.ctr@mail.mil, *Lockheed Martin*
Phillip G. Matthews, phillip.g.matthews2.ctr@mail.mil, *Lockheed Martin*
James C. Ianni, james.c.ianni.ctr@mail.mil, *Lockheed Martin*
Chris Sauerwald, csauerwa@altair.com, *Altair*
Thomas M. Kendall, thomas.m.kendall4.civ@mail.mil, *ARL DSRC*

Presenting author: Thomas M. Kendall, thomas.m.kendall4.civ@mail.mil, *ARL DSRC*
Primary Abstract Classification (Green Provisioning)
Primary Computational Technology Area/HPCMP Component - N/A

Abstract

This paper documents the early experiences and recent progress with employing the Energy-Aware Scheduler (EAS) at the DoD Supercomputing Resource Centers (DSRC). The U.S. Army Research Laboratory (ARL) has partnered with Lockheed Martin, Altair, and Instrumental to assess feasibility on current DSRC High Performance Computing (HPC) systems. Developmental work was completed on the ARL DSRC Test and Development systems and ported to the production systems at the ARL DSRC. The (EAS) is written in Python and works with the current program-wide scheduler, Altair PBS Professional, that is deployed across the DSRCs. EAS reduces power and cooling costs by intelligently powering off compute nodes that are not actively being used by the currently running or reserved for near future jobs. It has been estimated that the Energy Aware Scheduler could potentially save millions of Kilowatt-hours each year throughout the program. We will describe the extent of our work to date at the DSRC centers and our plans to complete our work by September 30, 2012.

Introduction

The HPCMP's Energy Aware Scheduler (EAS) project was initiated as a one year project under the HPCMP's Green HPC Initiative. The goal of the project is to evaluate the feasibility of reducing power and cooling costs, and the associated environmental impacts, that result from the energy used by idle compute nodes. IDC reports that the power and cooling costs for data centers surpassed the spending on new servers in 2007 and continues to grow as a fraction of data center spending [1]. Clearly, if the HPCMP is to continue to successfully meet the majority of requirements in the Department of Defense, efforts focused on reducing the energy costs need to be sustained.

Through the DoD High Performance Computing Modernization Program (HPCMP) the EAS was initiated to investigate and develop the ability to save energy by controlling power to the nodes of the various High Performance Computing (HPC) assets across the DoD Supercomputing Resource Centers (DSRC). This project is a collaboration between the U.S. Army Research Laboratory, Lockheed Martin, Altair and Instrumental. The HPC systems across all the centers run on an average 80 - 90% busy and it is attractive to reduce the power requirements on the portion of the systems that are idle. This capability is architecture specific and has been deployed on several systems throughout the DSRCs and has proven to be reliable and effective in reducing power consumption. The team tested the resiliency and feasibility of EAS on SGI Altix ICE, CRAY XE6 and Appro Xtreme (Utility Server architecture) by running benchmarks on these systems with the results published later in this paper. EAS works with

the current program wide scheduler, Altair PBS Professional™ Scheduler, by powering off resources (nodes) that are idle, to save energy. Typically idle nodes consume about 50% as much energy as active nodes and it is attractive to power off resources that are idle to save energy. It is projected that across the entire HPCMP program there is a potential to save millions of kilowatt hours per year. Annual estimated HPCMP kWh savings as a function of the idle percentage and number of hours per day that nodes could be made available for powering off are shown in Table 1.

Architecture	*10% 12h/D*	*15% 12h/D*	*10% 24h/D*	*15% 24h/D*
Cray XE6	*224,012*	*336,018*	*448,024*	*672,036*
SGI Altix ICE	*263,725*	*395,588*	*527,450*	*791,175*
Subtotal	*487,737*	*731,606*	*975,474*	*1,463,211*
Site (1.8 PUE)	*390,190*	*585,284*	*780,379*	*1,170,569*
Total	***877,927***	***1,316,890***	***1,755,853***	***2,633,780***

Table 1. Annual kWh Saved

We have integrated the Energy-Aware Scheduler (EAS) with Altair PBS Professional™ to control and minimize the power of resources that are not in use. Altair's PBS Professional™ Scheduler already incorporated a capability that was oriented toward power monitoring and control. After performance testing on ARL DSRC Test & Development System (TDS) the original Perl version of the main tool showed that it took at least 20 minutes to complete a scheduler cycle for a 1,000 node system. The script spent a lot of time trying to predict what the scheduler would do, and this consumed considerable time. An additional problem was that the predictions were based on the scheduler's most basic behavior and did not do sorting formula, backfill, strict ordering, node sorting, placement sets and many things that we can expect schedulers in the program to use. Beyond being slow, it may well have been inaccurate for the complexity of the DSRCs. Altair suggested that a Python implementation would be better at delivering on the requirements of EAS. The new version takes the approach of focusing on what the PBS Professional scheduler thinks will happen in the future, rather than trying to make the determination itself. This leaves all the scheduling logic with the scheduler and will adapt as configuration changes are made to the schedule. The only exception is the very basic mechanism for node sorting added for the 'keep minimum nodes online' feature at the tail end of the project to better support interactive and debugging requirements, especially on the Utility Servers.

Results

ARL DSRC SGI Systems

After the initial Python implementation was completed by Altair, the ARL DSRC team began testing the EAS code on a 96 core SGI Altix ICE Test & Development System (TDS) named icecube and saw significant improvements. While analyzing EAS on icecube several additional features were developed to enhance the code, to include "live" power lock file support that prevents race conditions and a minimum delay between scans to prevent it from running too often. Scripts specific to the SGI Altix ICE were constructed to power up/down nodes with calls initiated through EAS to manage the resources. EAS has 3 modes of operation, passive simulation, active simulation and live power. "Passive simulation" mode executes scripts to make changes to node resources to simulate powering nodes on/off but do not effect workload scheduling. "Active simulation" mode executes scripts to make changes to node

resources to simulate powering nodes on/off and effect's workload scheduling to give a realistic picture. "Live power" mode executes scripts to make all normal resource changes from the other nodes and issues the real commands to power nodes on and off. An additional command utility was built named "pstat" to display a snapshot of the EAS power state of nodes currently being managed by the power management scripts:

Name	***Power State***	***Marked idle/Powered off at***	***PBS state***
R1i3n8	*Powered-off*	*OFF: Fri Apr 6 16:35:24 2012*	*down*
R1i0n0	*Powered-off*	*OFF: Fri Apr 6 16:35:24 2012*	*down*
R1i0n1	*Powered-off*	*OFF: Fri Apr 6 16:35:24 2012*	*down*
R1i1n1	*Powered-off*	*IDLE: Fri Apr 6 16:15:16 2012*	*free*
R1i0n8	*Powered-off*	*IDLE: Fri Apr 6 16:20:53 2012*	*free*
R1i0b9	*Powered-off*	*IDLE: Fri Apr 6 16:20:53 2012*	*free*

Table 2. pstat

In early December 2011, we implemented "live power" mode on the TDS SGI Altix ICE, icecube, and tested extensively before employing on the production system. In mid December, EAS was configured on Harold, a 10,752 core SGI Altix ICE 8200 system, and began testing in "passive simulation" mode. Following 2 weeks of significant testing on the cluster, we turned "live power" mode on, one rack at a time, starting with the racks dedicated for reservation jobs only. We soon discovered problems on Harold due to a bug where Altair code was dereferencing a pointer inside the python binding that should not have been dereferenced. This eventually lead to the crash of the pbs servers. This issue returned often and manifested when the pbs server daemon restarted it's internal python interpreter. This problem was fixed in 11.x era and went away when 11.2 was installed on harold. Also, in 10.4.7+ there is a bug with the Python 2.5.1 distribution in the pwd module. Any hook using the pwd module could have a similar dereference issue as the first bug, but it occurred less frequently. The pwd module was used inside the old SLM/PBS integration hooks only, so removing them fixed the issue. (New versions of these hooks don't use pwd any more). To mitigate these PBS crashes, EAS was configured to only run 18:00 and 08:00 daily.

In early February 2012, Harold was upgraded from PBSPro 10.4.7 to 11.2 and EAS was configured for a second time in "live power" mode 24x7. Perl scripts were developed to generate accounting report details on power cycled nodes and total number of kilowatt hours saved by EAS. In addition, benchmarks were run on the system before and after the implementation of EAS and these demonstrated no significant performance degradation in job execution times. The results of these benchmarks are published later in this paper. Testing by SGI determined the SGI Altix ICE draws 147 watts from each idle node. From January to February 2012, we saved a total of 6,910 kWh on Harold by powering down idle nodes. Since Dec. 1, 2011 there have been 4 nodes failures (3 BMC, 1 HCA) as a result of EAS. Perl scripts were developed to generate reports on the number of nodes power cycled and the total number of kilowatt hours saved by EAS. Savings recognized since February on HAROLD are in the table below. As Harold is in the middle of its expected lifecycle, the available idle time is not as high as is typically experienced on HPCMP systems in the early and late stages of their lifecycles. It is expected that the savings will significantly increase as Harold approaches and enters year four of its lifecycle.

Dates	***EAS hours saved***	***EAS Hours saved x 147w***
02/06 - 02/12	*3971*	*583 kWh*

02/13 – 02/19	*4282*	*629 kWh*
02/20 – 02/26	*5240*	*770 kWh*
02/27 – 03/04	*2597*	*381 kWh*
03/05 – 03/11	*9084*	*1335 kWh*
03/12 – 03/18	*2331*	*342 kWh*
03/19 – 03/25	*6370*	*936 kWh*
03/26 – 04/01	*1116*	*164 kWh*
04/02 – 04/08	*1595*	*234 kWh*
04/09 – 04/15	*1360*	*199 kWh*
04/16 – 04/22	*3734*	*548 kWh*
04/23 – 04/29	*1780*	*261 kWh*
TOTAL		***6388 kWh***

Table 3. Harold kWh saved

During weekends, Harold typically has additional nodes available for power-down, as most of the eligible "backfill" type jobs have completed. The backfill jobs have wallclock time of less than 24 hours, and can also run on the reservation nodes of the cluster. When 'large' jobs are submitted, the scheduler will try to keep nodes idle in order to prevent backfill jobs from pushing the top job start time out. While these nodes are idle, they are subject to being shut down by power management to reduce wasted energy. Typically jobs on Harold are 64 nodes or less. Large jobs are typically 256 nodes or higher.

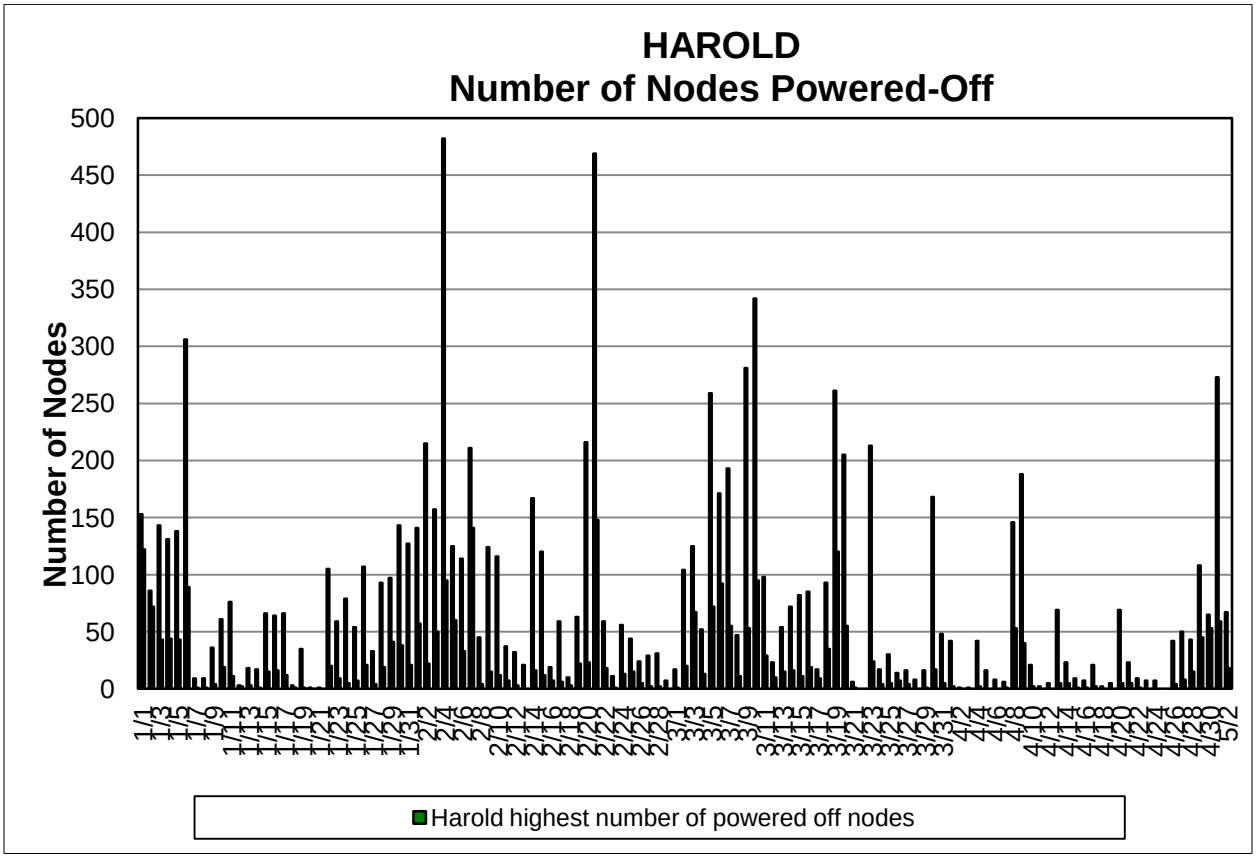

Figure 1. Harold

In early March, EAS was configured on a second SGI Altix ICE system, TOW, a 6,656 core SGI Altix ICE 8200 system. TOW was first configured in "passive simulation" mode and then slowly transitioned over to "live power" mode, one rack at a time. On this system, we saved 9,893 kWh in the month of March alone by powering down idle nodes. TOW runs a significantly different job mix than Harold and after a month of running "live power" mode we started to notice a trend in jobs randomly dying on startup. EAS was shutdown to rule out the power monitoring process and jobs returned to running as expected, without dying at start up. The problem was diagnosed to be associated with a lustre issue when starting nodes. SGI is investigating the problem, and has implemented a lustre patch to improve the reliability of node start-up. Energy savings with EAS on TOW are in the table below.

Dates	***EAS hours saved***	***EAS Hours saved x 147w***
03/03 – 03/04	*46*	*6 kWh*
03/05 – 03/11	*3614*	*531 kWh*
03/12 – 03/18	*6424*	*944 kWh*
03/19 – 03/25	*11810*	*1736 kWh*
03/26 – 04/01	*2983*	*483 kWh*
04/02 – 04/08	*3214*	*472 kWh*
04/09 – 04/15	*0*	*0 kWh*
04/16 – 04/22	*0*	*0 kWh*
04/23 – 04/29	*0*	*0 kWh*
TOTAL		***4172 kWh***

Table 5. TOW kWh saved

On TOW, there is a significant opportunity for potential power savings, mainly due to the large core count jobs that run on this system. Since the large jobs typically take time to accumulate enough nodes to run, we have a large amount of nodes that can be powered off. Also, given that new jobs are typically not submitted during this weekend on this system we have additional idle nodes that are candidate for power-down as can be seen in figure 2.

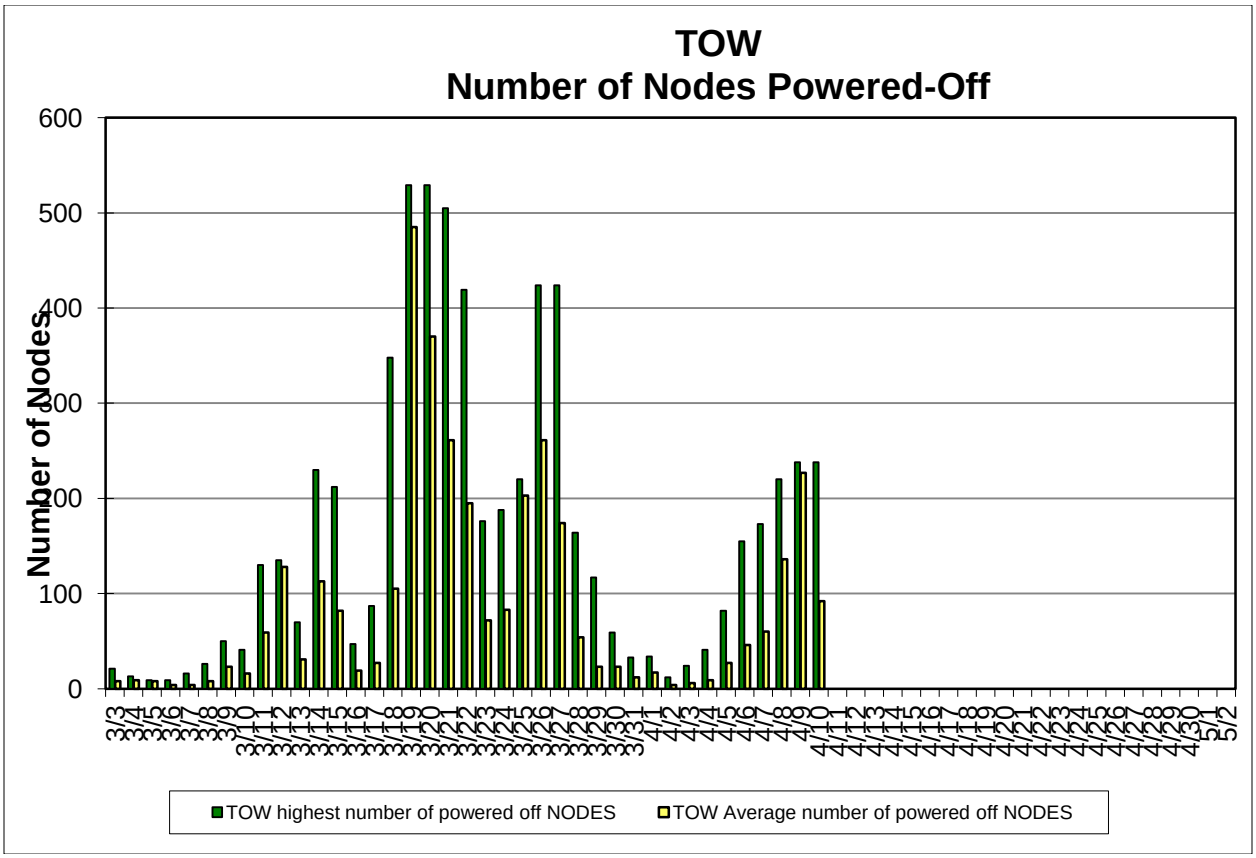

Figure 2. TOW

MHPCC DSRC Dell System

During the week of March 12, 2012, Chris Sauerwald, Altair and Bill DeSalvo, Instrumental, visited MHPCC to configure EAS on both Mana, 9,216 core Dell PowerEdge system, and the Utility Server. EAS was configured in "active simulation" mode. Since reservations are handled differently on these systems, Altair built a special PBS version, **11.2.2**, to support Maui's requirement for reservations that allows reservations to cross dedicated time boundaries. Because of the large percentage of nodes in use by reservations, a feature was added to EAS that allows any reservation longer than X seconds to have their nodes powered on ONLY if jobs need them. If reservations are less than X, their nodes will be powered on in time for the reservation start. From March 13-31 the virtual savings on Mana were 203,711 hours for a total of 30,556 kWh and a dollar savings of approx $11,000 at $0.37/kWh. From April 1-17 there were 261,207 hours of virtual savings. Mana was enabled in "live power" mode on April 25 and during the first day alone, there were 935 nodes powered off by EAS, about 80% of the system. The estimated dollar savings are $25,000 for the month of April

Utility Servers

We have integrated EAS on 3 out of 6 Utility Servers across the DSRCs, 2 at ARL DSRC and 1 at the MHPCC DSRC. The Utility Servers have a requirement for a given number of nodes to always be powered-on and available for user jobs. Subsequently, Altair updated the EAS code to support this requirement. The Utility Servers are fairly new to the program and as such not highly utilized, there is an excellent opportunity to power down idle nodes. The early numbers suggest a savings of about 27,000 hours a month on the unclassified Utility Server. On the unclassified Utility Server there is a good balance between powered-off nodes, and current workload on the system as can be seen in Figure 3.

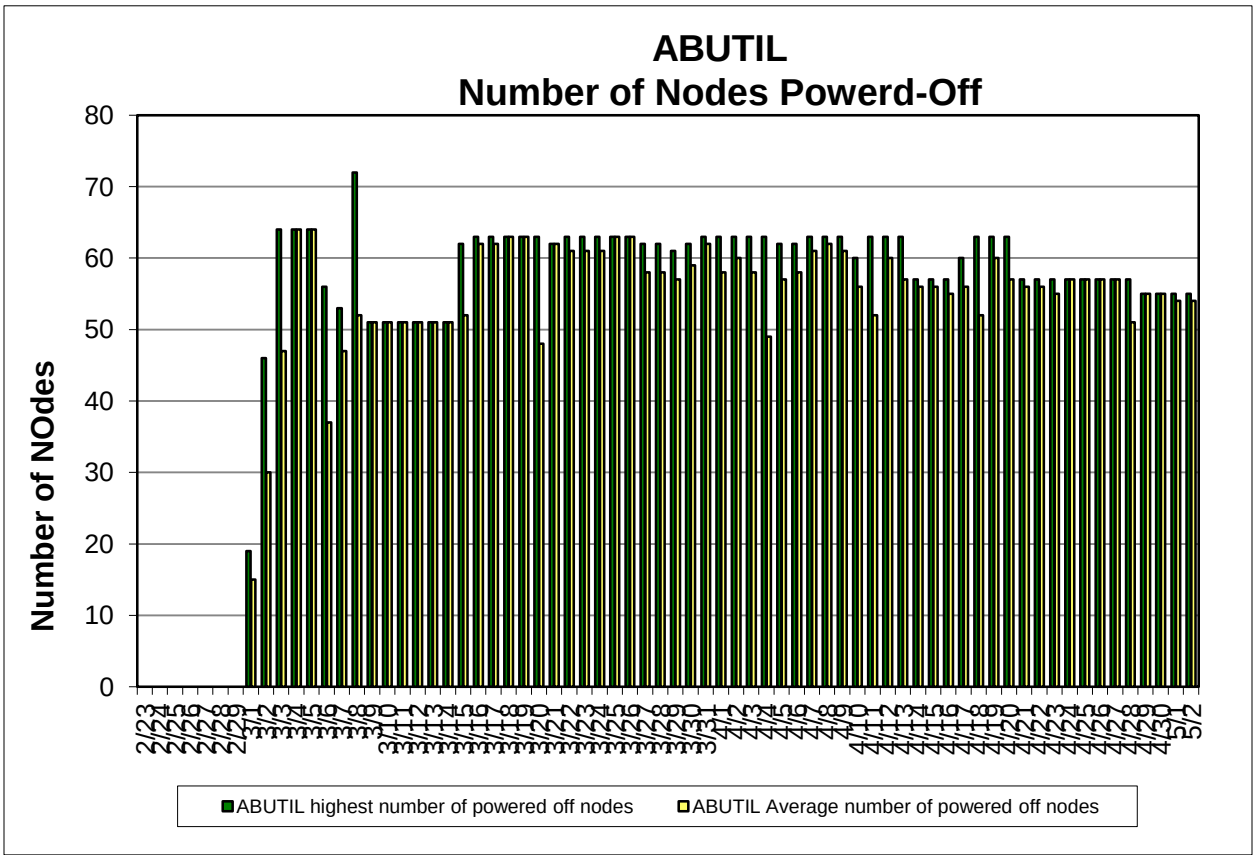

Figure 3. Abutil

We have begun to integrate the classified Utility Server with EAS. The graph in figure 4, displays how we have gradually added nodes, ran for several weeks, and upgraded the system on Apr 11, 2012, then disabled EAS to allow for further application testing of the cluster by the support staff. We plan to enable EAS on only the weekends in order to provide for uninterrupted user access during the weekdays until we can install the 0330 version of EAS.

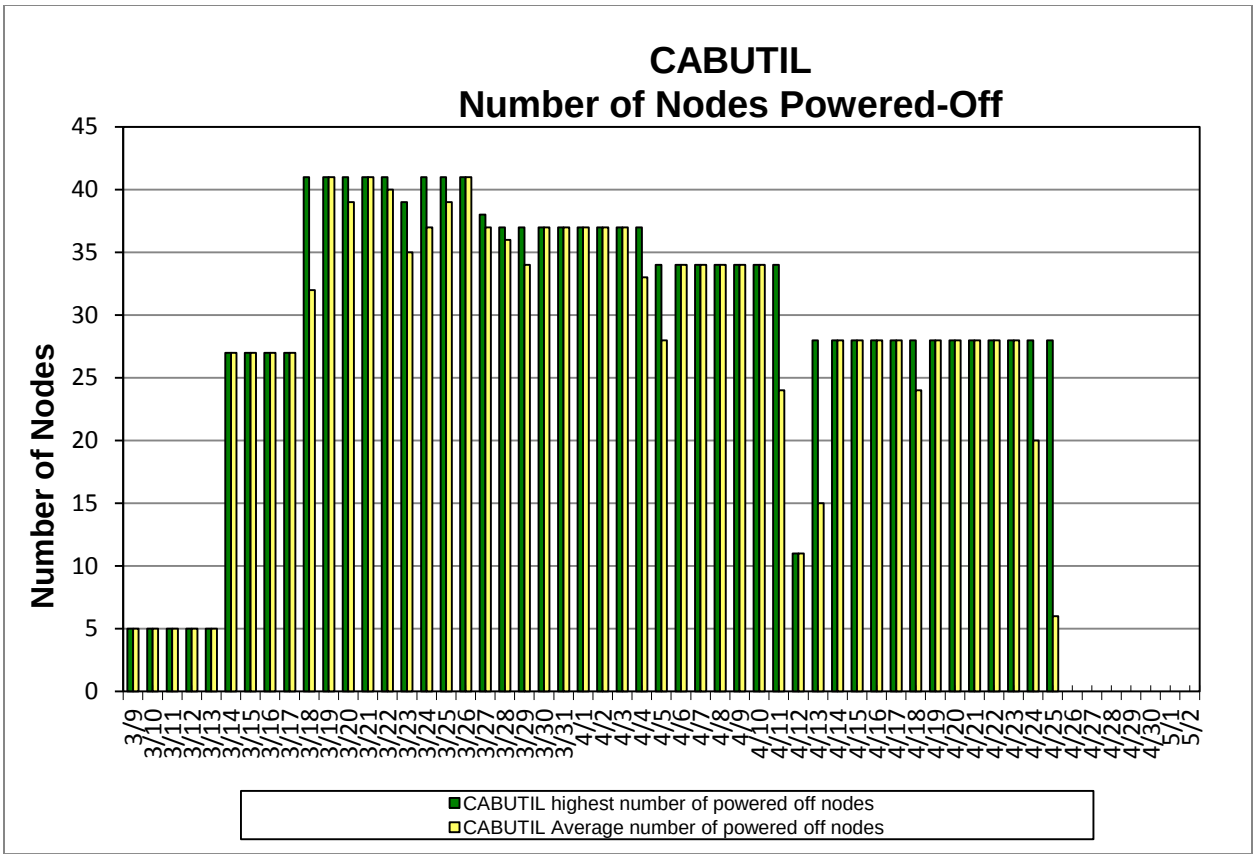

Figure 4. Cabutil

CRAY XE6

We began to investigate the potential cost savings using EAS on a CRAY XE6. The early testing revealed promising results. On a 16 compute node CRAY XE6, Tana, at the Arctic Region Supercomputing Center (ARSC), Dr. James C. Ianni, Lockheed Martin, and Liam Forbes, University of Alaska Fairbanks, tested compute node power utilization and the possible impacts of idle nodes shutting down and booting during running jobs. The first test was to shut down all idle compute nodes while an 8 node GAMESS job was running. The job was configured to use half the cores and all memory on each node. The test was performed 3 times. The second and third execution completed successfully, however, on the first execution, the running job ended after running only 7 minutes 50 seconds with an exit status of 0. The cause of the job failure was unable to be determined. The next test case was to boot 8 compute nodes while the same GAMMESS job was running. This test was also executed three times with all tests completing successfully. The job run times during all tests did not vary substantially, except for the one case. On Tana, we don't believe we are testing at a scale to demonstrate possible impacts on job run times and are working to test larger cases and/or different applications on one of the larger CRAY XE6s in the HPCMP to investigate possible impacts. Other similar tests were performed to get baseline measurements and to try powering down larger portions of the system (for example a blade). Based on those tests, we concluded that if EAS is implemented on a CRAY XE6, the focus should be on manipulating individual compute nodes. Trying to manipulate blades, cages, or cabinets "breaks" the system interconnect, either killing running jobs or crashing the system. Liam Forbes collected power measurements during all tests and calculated that a CRAY XE6 compute node draws approximately 115 watts at idle. There is minor hardware differences between the XE6 compute nodes at the different DSRCs, so some minor verification tests should be run on each system, but we think this is a reasonable estimation for calculating possible cost savings based on average numbers of idle nodes on each system. Discussions with Cray Inc. support and engineering regarding impacts and possible mechanisms to implement EAS are ongoing.

Benchmarks

Benchmarking can be a measure of any possible degradation of the systems network during a computational node disruption through power cycling. The DSRC has a dedicated suite of software (the Sustained Systems Performance suite or SSP) to provide an adequate benchmarking test on large superclusters. The Energy-Aware Scheduling (EAS) system was installed on a test system. Due to the small size of the test systems, the full SSP system was not utilized. Due to this constraint an alternative to the SSP benchmarking suite was utilized.

Those alternatives were SKaMPI [1] and a custom input to the GAMESS computational chemistry system [2]. SKaMPI is a benchmark for testing the MPI communications system and can do measurements of point to point communication, collective communication MPI operations and many others. Most of the MPI calls with various sized send/receive MPI buffers are called. The total time for the entire SKaMPI suite run is recorded as one benchmarking time. The GAMESS run for this benchmark utilizes the second order energy correction of Moller-Plosset perturbation theory (MP2) with the molecule benzoquinone. A geometry optimization calculation was performed on the benzoquinone. The basis set employed is Pople's double-zeta basis with an additional d-polarization function on the heavy atoms (6-31G*).

The SKaMPI Benchmarks were initially run on Harold before and after the installation of EAS on a subset of Harold. The runs were performed on 64 cores (8 nodes, 2 GB/core) under OpenMPI-1.4.1. As shown below, there was a positive significant change in the timings of the SKaMPI runs. This is probably not due to the installation of EAS, but the reconfiguring of Lustre to use MPI-I/O locking between the December 14th and January 10th time period.

Machine	Run Date	Start	End	Elapsed Time
Harold				
	Wed Dec 14 18:48:11 EST 2011	6:48:11	8:03:23	1:15:12
	Tue Jan 10 12:57:11 EST 2012	12:57:11	13:40:58	0:43:47
	Thu Jan 12 12:35:59 EST 2012	12:35:59	13:25:43	0:49:44
ARL Utility Server				
	Wed Dec 14 15:56:39 CST 2011	15:56:39	17:17:17	1:20:38
	Tue Jan 10 13:06:17 CST 2012	13:06:17	14:32:56	1:26:39

Table 1.

The SKaMPI Benchmarks were also run on the ARL Utility Server before and after the installation of EAS. The sam benchmarks were also executed on a small subset of Harold. The runs were performed on 64 cores (8 nodes, 2 GB/core) under OpenMPI-1.4.1. As shown in Table 1, there were no significant changes in the timings of the SKaMPI runs.

A decision was made to switch to a more realistic test case for the ERDC machines. In this case, the above mentioned GAMESS system was executed. Although the EAS system was not installed on the ERDC test system for Chugach, named Tana, benchmarks were run before and after nodes were power cycled to simulate what the EAS system would perform. Those runs are shown in Table 2. The first two runs are the reference runs where none of the nodes in the entire Tana test system were power cycled. Runs 3 to 16 represent a series of runs where Tana nodes

were power cycled. With the exception of run 11, most runs did not deviate too far from the 1,059 second reference runs. The average and standard deviation of those runs are 1,062 +/- 4 seconds indicating almost consistent timings. The outlier at run 11 could be attributed to a transient network error in the Tana test system when that job ran. The output had a "CqWaitEvent failed in Wait; err 11" in the output. This network error comes from Cray's GNI (General Network Interface) CqWaitEvent() function and not from MPI or from the GAMESS code. This error was not repeatable.

Run #	GAMESS Run (sec)
1	*1059*
2	*1059*
3	1064
4	1061
5	1059
6	1070
7	1058
8	1065
9	1057
10	1068
11	466
12	1063
13	1059
14	1062
15	1057
16	1064

Table 2.

Significance to DoD

The Energy-Aware Scheduler has proven to be feasible and reliable on the different architectures that we have tested and measured powered savings are encouraging.

Conclusions

In this paper, we gave a brief overview of the efforts to deploy the Energy-Aware Scheduler across the HPCMP. Implementing the Energy-Aware Scheduler on the ARL DSRC systems has thus far saved 1,000's of kWh and also has the potential to save in reducing cooling requirements, with very little fallout from hardware failures. We are in the process of working with the other DSRCs and Original Equipment Manufacturer vendors to further expand this capability. We have heard many concerns while implementing EAS that we are working to address These concerns range from the number of times a blade can be powered up/down, system stability; increase in System Administrator workload due to troubleshooting increased on the nodes that are powered off. With this enhanced power efficiency we can lower costs at DoD centers. Early results have proven that there is the potential to save the projected millions of kWh throughout the HPCMP program. This project is scheduled to be completed in September 2012.

Acknowledgements

This effort was supported in part by the HPCMP's Green Initiative.. The authors would like to thank Liam Forbes for all his research that he did on Tana. We would also like to thank MHPCC for their work in implementing EAS. We look forward hearing your success stories with EAS.

References

[1] IDC Report "The Impact of Power and Cooling on Data Center Infrastructure.http://www-03.ibm.com/systems/cn/resources/systems_cn_IDC_ImpactofPowerandCooling.pdf

[2] SKaMPI, Werner Augustin and Thomas Worsch, April 7, 2008 version, http://liinwww.ira.uka.de/~skampi/ .

[3] GAMESS, Version 24 Mar 2007 (R6), Iowa State University, M.W.Schmidt, K.K.Baldridge, J.A.Boatz, S.T.Elbert, M.S.Gordon, J.H.Jensen, S.Koseki, N.Matsunaga, K.A.Nguyen, S.J.Su, T.L.Windus, M.Dupuis, J.A.Montgomery, J.Comput.Chem. 14, 1347-1363(1993)